Chapter 5

Protection of Relational Databases by Means of Watermarking: Recent Advances and Challenges

Javier Franco Contreras and Gouenou Coatrieux

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.68412

Abstract

Databases represent today great economical and strategic concerns for both enterprises and public institutions. In that context, where data leaks, robbery as well as innocent or even hostile data degradation represent a real danger, and watermarking appears as an interesting tool. Watermarking is based on the imperceptible embedding of a message or watermark into a database in order, for instance, to determine its origin as well as to detect if it has been modified. A major advantage of watermarking in relation to other digital content protection mechanisms is that it leaves access to the data while keeping them protected by means of a watermark, independent of the data format storage. Nevertheless, it is necessary to ensure that the introduced distortion does not perturb the exploitation of the database. In this chapter, we give a general overview of the latest database watermarking methods, focusing on those dealing with distortion control. In particular, we present a recent technique based on an ontological modeling of the database semantics that represent the relationships in between attributes—relationships that should be preserved in order to avoid the appearance of incoherent and unlikely records.

Keywords: watermarking, relational database, information security

1. Introduction

The evolution of multimedia technologies and communications has resulted in a remarkable increase in the construction, transfer and sharing of databases. As a consequence, data gathering and management into databases or data warehouses or the scaling up to big data become important economical and strategic concerns for enterprises and public administrations in decision making. The expansion of data-mining and assisted analysis tools are just

two examples that highlight the growing value of these databases. In that context, information leaks, thefts (confidentiality, traceability) or even degradations (integrity/authenticity), intentional or not, represent a real menace. This has recently been proved by the Wikileaks [1] or the Falciani cases [2], where large amounts of sensitive data have been exposed publicly on the Internet due to internal leaks.

Several protection mechanisms have been proposed so as to protect digital contents. A nonexhaustive list encompasses user authentication, access control and encryption which are helpful for confidentiality, digital signatures that can support data integrity and non repudiation and logs that can help to trace data. However, these security solutions offer an *a priori* protection in the sense that once they are bypassed, or more simply when the data access is granted, data are no longer protected.

On the contrary, watermarking can complementarily provide an *a posteriori* data protection. By definition, watermarking lies on the insertion of a message (some security attributes) into a host document (e.g., an image, an audio signal or, in our case, a database) by slightly modifying it based on the principle of controlled distortion. Watermarking leaves thus access to the data, which can be manipulated or consulted, while staying protected by means of a watermark. This watermark or the message it corresponds to (or equivalently the embedded security attributes) may serve as the protection of the owner rights, data integrity, data traceability, etc. Its versatility makes watermarking a really attractive solution for sensitive data protection.

While there is vast knowledge in the field of multimedia watermarking [3, 4], the interest in database watermarking has been limited, to date, with about 100 publications since the seminal method of Agrawal and Kiernan, which dates to 2002 [5]. In particular, and as we will see in the sequel, relational database watermarking differs from multimedia contents watermarking in several points. Among them, two are worth highlighting—i) records in a database can be reorganized without changing the meaning of the database, in opposition to highly correlated neighbor samples in a signal or pixels in images and ii) the existence of specific manipulations a database may undergo like tuple suppression and insertion which will modify the database structure. At the same time, depending on the nature and on the sensitivity of the data, more or less strict distortion constraints have to be considered and managed or at least modeled.

This chapter addresses the latest advances on the protection of relational databases by means of watermarking. We focus, in particular, on methods that aim at preserving the informative content of database. If in the past distortion control techniques preserved database statistics, a recent one suggests taking into account the semantic meaning of database records.

This chapter is divided into five main sections. First, in Section 2, we present the main applications of database watermarking. In Section 3, we come back and sum up the basic principles of database watermarking, highlighting the main differences with watermarking of multimedia contents (i.e., images, video). Section 4 gives an overview of the existing database watermarking techniques, putting in evidence "how" distortion control in database watermarking is most of the time achieved. In this section, we describe, in more detail, a semantic distortion control

http://dx.doi.org/10.5772/intechopen.68412

by means of ontologies—a modeling that is much more general. This solution is illustrated considering a practical case with a medical database containing inpatient stay records.

2. Applications of database watermarking

As depicted above, watermarking stands for the insertion or dissimulation of a message (a watermark) into the records or the attributes' values of a database. Depending on the relationship between the host database and the embedded message, different applications have been proposed.

2.1. Copyright and ownership assertion

As in the case of multimedia content watermarking, the first developed and most-studied watermarking application corresponds to database copyright protection. It relies on the insertion of an identifier associating the host document to its owner (creator or buyer) [6]. This identifier or watermark should be imperceptible and resistant to any operations, especially those aiming at removing the watermark. The first database watermarking technique, introduced by Agrawal and Kiernan [5], focused on copyright protection.

2.2. Traitor tracing and database traceability

In some cases, the identification of the recipient of one database can be a priority so as to trace a possible illegal redistribution. Watermarking is referred in that context as "fingerprinting" [7]. Herein, each distributed copy of the content is marked with an identifier or fingerprint which uniquely identifies an individual. If one of the receivers decides to illegally reroute or redistribute the database, it becomes possible to identify him or her [8]. The way these fingerprints are built has received a lot of research effort in order to make them resistant to collusion attacks in which several users owning copies of the same content cooperate in order to obtain an unwatermarked version. Such fingerprints or user identifiers are anticollusion codes [9, 10] and have, as an objective, the identification of at least one or several colluders in a coalition of users.

In the same vein, such traitor tracing solutions can serve as the identification of a dishonest user at the origin of a data leak. As previously exposed, a message identifying the user is embedded when he/she accesses the content. If the information is retrieved online, it will be possible to identify the responsible person by extracting the message. Contrary to the previous problem, the collusion attack is of less concern as such data leaks are usually the result of one user.

2.3. Integrity control (tamper detection)

Integrity or authenticity control represents the third main application of database watermarking. Indeed, it is essential to ensure data integrity, especially when they acquire a legal value

or if they contribute to sensitive decision making. That is especially the case of the medical domain in case of litigations.

Fragile or semifragile watermarking constitutes attractive alternatives. In opposition to robustness, the fragility of the mark to databases' manipulations can herein be useful. The absence or the incorrect detection of a mark will indicate a loss of data integrity. Depending on the applicative context, the mark can be designed to resist some specific manipulations but not to all. If all manipulations have to be detected, we will talk about fragile watermarking [11, 12]. Such techniques are usually very sensitive, like a digital signature or message authentication code, and can indicate which parts of the database have been altered [13]. On the contrary, a semi-fragile watermark will be designed to be robust to some innocent manipulations, that is allowed in the applicative framework, and fragile to hostile attacks [14, 15].

3. Database watermarking: Specificities and a general chain of watermarking

A database *DB* is composed of a finite set of relations $\{R_i\}_{i=1,\ldots,NR}$. Hereon, for sake of simplicity and without loss of generality, we will consider one database based on one single relation constituted of N unordered tuples $\{t_u\}_{u=1,\ldots,N}$, each of M attributes $\{A_1, A_2,\ldots, A_M\}$. The attribute A_n takes its values within an attribute domain, and $t_u.A_n$ refers to the value of the n^{th} attribute of the u^{th} tuple. Each tuple is uniquely identified by either one attribute or a set of attributes, and we call its primary key $t_u.PK$. Tuples or attributes in such a database can be reorganized, removed and added by the user. In this section, we expose the fundamentals of how this kind of structure can be watermarked and with which purposes.

The application of existing signal or image watermarking techniques to databases is not a straightforward process. Relational databases differ from multimedia contents in several aspects that must be taken into account when developing a watermarking scheme.

3.1. Database structure and watermark insertion/detection synchronization

One of the main differences is that samples in a multimedia signal are sorted into a specific order, in a temporal (e.g., audio signal samples) and/or spatial domain (e.g., pixels of an image or video), giving a sense of the content itself to the user. Close samples are strongly correlated with usually important information redundancy. This is not the case of relational databases, the purpose of which is to provide efficient storage of independent elements within a common structure. Thus, tuples in a relation are not stored in any specific order. At the same time, because tuples or records can be stored and reorganized in many ways in a relation without impacting the database information, questions arise between the synchronization of the watermark insertion and the watermark reading/extraction processes. Indeed, with signals or images, one can count on their intrinsic structure, working, for instance, on blocks or groups of consecutive samples or on transformed coefficients so as to conduct the insertion of one symbol of the message. The same strategy is not so easy to apply in the case of relational

http://dx.doi.org/10.5772/intechopen.68412

databases where tuples can be displaced, added and removed. Identification of watermarked elements in a database (records or attributes) and consequently the synchronization between the watermark insertion and detection stages require specific solutions. In order to make the watermark insertion/reading independent of the database structure, or more clearly of the way this one is stored, a preprocessing step is usually applied before message insertion/reading and (see **Figure 1**) following different possible strategies.

The first approach [5] consists of secretly constituting two groups of tuples based on a secret key. One group contains the tuples to be watermarked while the tuples in the second are not modified. In order to obtain the group index of a tuple t_u in the relation R_I, it makes use of a HASH function (H) modulo a parameter $\gamma \in N$ which controls the number of tuples to modify. If we define $t_u.PK$ as the primary key of a tuple, K_S as the secret watermarking key, mod as the modulo operator and $\|$ as the concatenation operation, the condition $H(K_S \| H(t_u.PK \| K_S)) \bmod \gamma = 0$ indicates whether a tuple must be watermarked or not. In [16], the HASH operation is replaced by a pseudo-random generator initialized with the tuple primary key concatenated with the secret key. Notice that these methods allow for embedding a message of one bit only. This consequently restricts the range of possible applications. In order to increase the capacity, Li *et al.* [8] proposed an evolution of the previous method in which one bit of the message is embedded per selected tuple. To do so, the watermark bit to embed in the tuple t_u is also selected taking into account the tuple primary key $t_u.PK$ and the secret key K_s. This allows the insertion of a multi-bit watermark offering more applicative options.

A more advanced solution consists of a "tuple grouping operation," which outputs a set of N_g that is nonintersecting groups of tuples $\{G_i\}_{i=1,\ldots,Ng}$. This allows spreading each symbol of a message S (or equivalently of the watermark) over several tuples, increasing, then, the watermark robustness against tuple deletion or insertion (i.e., the capability to detect/extract the message even if the database is modified).

The first strategy proposed in [17] is based on the existence of special tuples called "markers" which serve as a boundary or frontier between groups of tuples organized in a user-dependent order. A group corresponds to the tuples between two group markers. More clearly,

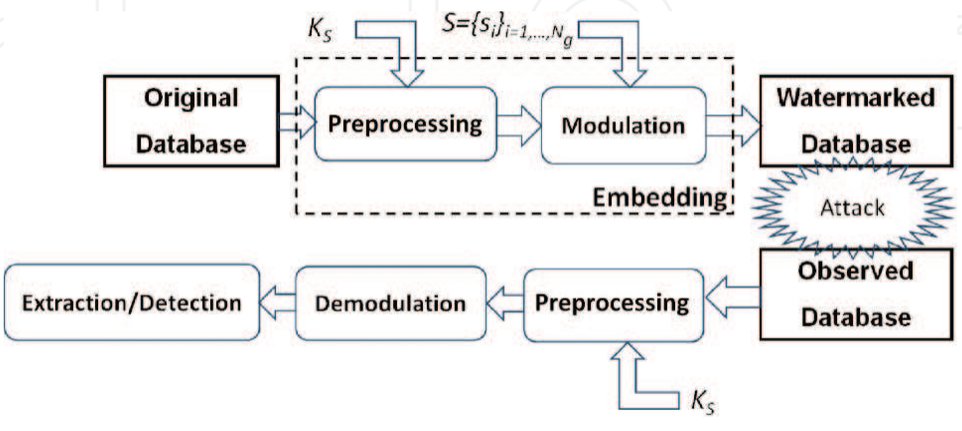

Figure 1. A common database watermarking chain. The message S is the concatenation of different symbols independently inserted into groups of tuples secretly constituted based on a secret watermarking key K_S.

tuples are ordered according to the result of a cryptographic HASH operation applied to the most significant bits (*MSB*) of the tuples attributes concatenated to a secret key K_S such as $HASH(Ks \| MSB \| Ks)$. Then, tuples for which $H(K_s \| t_u . PK) \bmod e = 0$, where e is a parameter that fixes the size of groups, are chosen as group markers. The main drawback of this approach stands on the deletion of some of the markers which will induce a loss of watermark symbols. The extracted message will be shorter than the one originally embedded. To overcome this issue, the most common strategy consists of calculating the group index number $n_u \in [0, N_g - 1]$ of t_u as in Eq. (1) [18]. Using a cryptographic hash function, such as the secure hash algorithm (SHA), ensures the secure partitioning and the equal distribution of tuples into groups.

$$n_u = H(Ks \| H(Ks \| t_u . PK)) mod Ng \quad (1)$$

Once this preprocessing task is conducted, one bit or symbol of the message is then embedded per group of tuples by modulating or modifying the values of one or several attributes according to the rules of watermarking modulation (e.g., modifying the attribute's statistics as in Ref. [17] or the tuple order as in Ref. [11]). Thus, with N_g groups, the inserted message corresponds to a sequence of N_g symbols—$S = \{s_i\}_i = 1,\ldots,Ng$.

Some other approaches that do not make use of the primary key for group construction have also been proposed. Shehab *et al.* [18] regroup tuples depending on the MSB of some attributes. The main disadvantage of this strategy stands on the fact that groups can have very different sizes as MSBs do not usually follow uniform distribution. In a similar way, and with the same disadvantage, Chang *et al.* [19] propose to construct a virtual primary key from a fragment of some categorical or textual attributes.

3.2. Database manipulations

Another important difference between multimedia and database watermarking is associated with the frequency and the nature of manipulations over the data. In the multimedia case, filtering and compression operations are common. They modify the signal samples' values but do not change the signal structure (a filtered image or of a part of it will be close to its original version). In databases, insertion and deletion of tuples are frequent. They can be seen as sub-sampling and oversampling operations but with irregular distribution, a quite rare situation in signal processing, especially if the process output should keep an image structure. Moreover, databases may be queried so as to extract pieces of information that present an interest to the user.

3.3. Numerical and categorical attributes

Beyond the database structure and manipulation, one must also consider that the information contained in a database may come from different sources, for example, different services in a hospital. Hence, the attributes of the database can be of very heterogeneous nature while having semantic logic. In particular, one may have to handle numerical and categorical attributes or complex data such as images and so on. Categorical attributes differ from numerical attributes in the absence of order relationships in between the values of their domain. For example, considering the attribute "*eye colour*," no rule states *a priori* that "*blue*" is greater or smaller than "*green*." It is then difficult to apply mathematical operations in this context. We cannot say what will be the result of "*blue*" plus "*brown*." This is not the case in multimedia signals where all the

http://dx.doi.org/10.5772/intechopen.68412

samples are numerical with the same dynamic. Nevertheless, solutions have been proposed to handle such categorical attributes even though it appears more difficult to control the distortion and preserve the meaningful value of the database. In a more general way, if image or video watermarking makes use of perceptual models of the human perception defects in order to mask the watermark, database watermarking requires other kind of distortion control solutions. As we will see, they are based on statistical and semantic aspects, some of which will be exposed in Section 4.

4. Overview of database watermarking schemes

This section presents an overview of the state of the art in database watermarking. Marking modulations are classified according to four criteria. The first two correspond to the robustness of the watermark or its fragility against database modifications. As stated earlier, robustness is the capability to retrieve the watermark after the protected database has been innocently (i.e., modifications that are authorized in the applicative framework) or malevolently (i.e., modification where the purpose is to remove the watermark) modified. Robustness is an important property in traitor tracing and copyright protection (ownership proof) frameworks. On the contrary, fragility of the watermark to some or all database modifications is a property that is at the basis of integrity control (tamper detection) applications.

The other two criteria are watermark imperceptibility and the database information the watermarking modulation uses so as to embed the watermark (e.g., categorical or numerical attributes, tuples' order, etc.). The former is a fundamental issue in database watermarking. This is why we propose a second classification level which depends on the way methods deal with data distortion. We will thus distinguish methods with or without distortion control, "distortion free" methods and lossless or reversible methods.

4.1. Robust methods

In the sequel, these methods are presented depending on the pieces of information they modulate in a database. We propose to distinguish three categories. *Distortion-based methods* modify or alter the values of some attributes of the database, these attributes being numerical or categorical, satisfying or not distortion constraints. The second class we suggest to consider regroups *lossless or reversible distortion-based methods*. The reversibility property ensures that it is possible to remove the watermark and to restore the original attributes' values of the database. The last class of methods modulates the database structure for message embedding. These schemes are referred to as *distortion-free methods*, due to the fact that they do not modify the record attributes values.

4.1.1. Distortion-based methods

4.1.1.1. Modification of numerical data

The first database watermarking method proposed by Agrawal and Kiernan [5] inserts a watermark by bit substitution into the least significant bits (LSB) of the database attributes' values. The tuples, attributes and bits to be modified are secretly selected by means of a hash operation

(see Section 3). In this scheme, the watermark bit sequence depends on the database content and is not known by the user, that is, it corresponds to a database "footprint." At the detection, if the database has been watermarked, the expected number of bit correspondences (or equivalently the correlation) in between the recomputed database footprint and the extracted watermark should be near to 100%, while this number logically falls down to 50% if it has not.

Li *et al.* [8] extended the previous method so as to allow the insertion of a sequence of bits: a multi-bit message. Considering thus a multi-bit message m, the jth bit of the tth attribute A_t of the record t_u, which we call b_j, is set to a value $b_j' = b \oplus m[q]$, where $b \in \{0, 1\}$ is a mask bit obtained from a random sequence generator S such as $b = S(K_S \| t_u.PK) \bmod 2$, $m[q]$ is the qth bit of m secretly selected based on a function of K_S and $t_u.PK$ and $\oplus$ is the xor operator. The message is inserted several times in the database. At the detection, for each secretly selected tuple, the operation $b \oplus b_j'$ is computed so as to extract the binary value inserted in $t_u.A_t$. This bit extraction is followed by a majority vote strategy so as to determine the final value of the extracted message. Such repetition increases the watermark robustness. Since Li *et al.*, different approaches following the same embedding strategy have been proposed with as objective to increase the complexity of message extraction or tampering an attacker [20, 21].

4.1.1.2. Modification of categorical data

Categorical attributes differ from numerical data in the absence of order relationships in between the values of their attribute domain. Sion *et al.* [22] were the first to propose a method for this kind of data. Let us consider an attribute A_t which takes its values in the finite value domain $\{a_1, a_2, a_3, \ldots, a_{Na}\}$. These different values do not have a predefined order. However, a numerical value can be arbitrarily assigned to each categorical value creating thus a virtual attribute dynamic as for numerical attributes. By doing so, they can then apply a numerical attribute modulation, for instance LSB substitution. The main problem of this method is the strong distortion it can introduce when the meaning of the new value is considerably different from the original one.

4.1.1.3. Introduction of "fake" tuples and/or attributes

Another type of method is based on the insertion of new pieces of information (e.g., tuples or attributes) into the database. In that case, even though the original information has not been modified, one can consider that a certain distortion results from the additional data. Indeed, they can bias the result of database queries or of some statistical analysis. Pournaghshband [23] presents a method that inserts false tuples. In order not to impact the database integrity or coherence, it constructs primary key values for each new tuple so as to respect the key integrity constraint (there should not be duplicated primary key values). The detection seeks for fake tuples. The presence of one of them indicates that the database has been watermarked as they are only known to the database owner.

In a scenario not too different of this one which focuses on the watermarking of ontologies, we find the method of Suchanek and Gross-Amblard [24] based on the same strategy of false information insertion in order to identify the ontology owner.

4.1.2. Distortion control-based methods

In order to increase the watermark imperceptibility and to not modify the normal use of the data, distortion control techniques have been considered. All of them work on numerical attributes. Gross-Amblard published in 2003 a theoretical work [25] oriented to distortion minimization in the case of *a priori* known aggregation queries. Minimal distortion is considered to be obtained if the result of these queries is exactly the same as the one obtained with nonwatermarked original data. In this framework, Gross-Amblard modulates pairs of tuples involved in the result of the same query with distortion of identical amplitude but of opposite sign for each tuple in the couple so as to compensate introduced perturbation in average. This algorithm has been extended and implemented in the Watermill method proposed by Lafaye *et al.* [26]. A limitation of this approach is that queries should be *a priori* known. Moreover, only aggregation queries are considered. Regarding other kind of queries (e.g., selection of a set of tuples), Lafaye *et al.* apply the method of Sion *et al.* [17] which is based on the modification of attribute's values statistics under information quality constraints, defined by means of the mean squared error (MSE). Once groups of tuples are constructed, Sion *et al.* compute a reference value *ref* that is calculated in each group according to the mean (*avg*) and the standard deviation (σ) of the attribute to the watermark such as: $ref = avg + c\sigma$, where $c \in (0, 1)$ is a user-defined parameter. The embedded bit value depends on the number of attributes' values in a group v_c that are over this reference. More clearly, for a group of Nt tuples, insertion relies on two parameters, v_{true}, $v_{false} \in (0, 1)$, in a way that a bit "0" is embedded if $v_c < Nt \times v_{false}$ and a bit "1" is embedded if $v_c > Nt \times v_{true}$. At the same time, if the modification exceeds the quality constraints, a fixed threshold or a rollback operation is applied, that is, all the operations performed onto the tuples of a group are undone.

Shehab *et al.* [18] enhanced the method of Sion *et al.* with a more efficient management of distortion constraints, while solving, at the same time, some issues linked to the group creation strategy (see Section 2.2). Watermarking is presented as a constrained optimization problem, where a dissimulation function Θ is maximized or minimized depending on the bit value to embed. The optimization space is limited by the quality constraints set. In the example given by the authors, Θ represents the number of elements which exceed a certain reference value (same value as in the method of Sion *et al.*). At the detection, the value of Θ is calculated and the detected bit is a 1 (resp. 0) if its value is greater (resp. smaller) than a threshold T. The value of T is calculated from the embedding information so as to minimize the probability of a decoding error.

Lately, Kamran *et al.* [27] have proposed the concept of "once-for-all" usability constraints. Considering a database that should be transferred to several users, they proved that if the detection threshold is fixed in order to ensure a correct detection for the most restrictive set of constraints, then detection reliability is independent of the constraints. This most restrictive set of constraints can be named "once-for-all" usability constraints. One drawback of this method is that its robustness and the lowest distortion it induces stand on a very short mark embedded into a few number of tuples. If the "good" tuples are altered, that is the watermarked tuples, the database is unprotected. Notice also that the modulation on which this scheme is based has some security issues allowing detecting any chosen watermark even if it has not

been embedded into the database. The same authors propose in another work a watermarking scheme that preserves classification results of an *a priori* known data-mining process [28]. To do so, attributes are first grouped according to their importance in the mining process. Then, some local (i.e., for a set of attributes) and global constraints are derived from some dataset statistical characteristics that are relevant to the mining process. Finally, the allowed perturbation for a set of attributes is determined by means of optimization techniques.

As it can be seen, all the above methods aim at preserving statistical properties of the database. They do not consider the existence of strong semantic links in between attributes values in a tuple, links that should be preserved when modifying attributes values. Indeed, tuples must remain semantically coherent in order to: (i) assure the correct interpretation of the information without introducing incoherent or unlikely records and (ii) keep the introduced perturbations invisible to the attacker. In order to solve these issues, Franco-Contreras and Coatrieux propose to consider an ontological modeling of the semantic relations between attributes values in the database so as to guide the watermark embedding [29]. Being the most recent approach in dealing with attributes distortion control, we will present it in more detail in the next section.

4.1.3. Lossless or reversible methods

4.1.3.1. Lossless watermarking of numerical data

In some cases, there is an interest or even a need of being able to recover the original database from its watermarked version. For instance, one may want to perform some operations on the original data or update the watermark. The reversibility property is herein of great interest. Robust lossless watermarking has been recently considered in the context of relational databases. Most of the existing methods are an adaptation of techniques proposed for image watermarking [30] and, as these, they are predominantly fragile with some exceptions.

Let us start by the latter, that is, robust methods. In Ref. [31], Gupta and Pieprzyk propose a zero-bit watermarking method where a binary meaningless pattern is embedded into secretly chosen tuples with attributes which are real numbers. To do so, a secretly chosen LSB from the integer part of an attribute is replaced by a pseudo-random-generated bit. To make the scheme reversible, the original LSB value is inserted into the space left by shifting the LSB representation of the fractional part of the attribute. The presence of the binary pattern is checked by the detector, indicating if the database has been watermarked or not. In order to reduce data distortion, Farfoura *et al.* [32] suggest watermarking the fractional part of one numerical attribute by means of prediction-error expansion modulation originally proposed by Alattar in [33] for images. Although this method is robust against common database manipulations (e.g., tuple addition or removal), the watermark will not survive a simple rounding integer operation. Beyond, it is important to notice that difference expansion modulation has not been designed for being robust to attributes' values modifications (this is the same for images). Indeed, Farfoura *et al.* [32] achieve watermark robustness with the help of a majority vote strategy, repeating thus several times the message into the database.

On its side, the method by Li *et al.* [34] constructs groups of tuples according to a clustering technique. The maximal modification that can be introduced into a tuple ensures that it will remain in the same group from the detector point of view. The watermarked value of an attribute is then calculated from an expansion of the polar angle of the attributes to the watermark. However, and as reported by its authors, this method is not fully reversible as some little errors can be found in the recovered data.

Recently, Franco-Contreras *et al.* [35] adapted the lossless watermarking scheme based on circular histogram modulation, originally proposed for images by De Vleeschouwer *et al.* [36], to the watermarking of relational databases. More precisely, this scheme modulates the relative angular position of the circular histogram center of mass of one numerical attribute in the relation. This scheme allows the embedding of a robust sequence and a fragile sequence at the same time and it can be thus considered for ownership control and traceability as well as for integrity control. Details on experimental and theoretical performance of this scheme can be found in Ref. [35].

4.1.3.2. Reversible watermarking of categorical data

In the method proposed by Chang *et al.* [19], one bit of a message is embedded by replacing the last letter of the last word of a textual attribute with another one from previously constructed reference sets. More precisely, before message embedding, two reference sets are constructed, one for each possible bit value, "0" or "1," which simply correspond to a secret ordering of letters in the alphabet, that is {*a*, *b*, …, *z*}. At the detection, the knowledge of these reference sets allows for extracting the embedded bit as well as the restoring of the original letters' values. If high robustness against classic attacks is achieved, the use of a spelling checker will help erase the embedded message.

4.1.3.3. "Attribute distortion-free" methods

In the above methods, it is assumed that a slight distortion can be carried out for message insertion without perturbing the interpretation or any *a posteriori* uses of data. However, if one may consider that no data perturbation can be introduced, methods that do not modify attributes values can represent as interesting alternatives. These attribute distortion-free robust embedding strategies play on the way textual or categorical attributes values are encoded.

Al-Haj and Odeh [37] embed a binary image by modifying the number of spaces between words. In the same vein, Hanyurwimfura *et al.* [38] take advantage of the Levenshtein distance between words in order to select the words between which the space can be modified, those at the smaller distance. **Figure 2** illustrates the application of this modulation on a textual attribute. It is considered that such kind of modification does not induce any information quality loss. Instead of modifying the spaces in between words, Shah *et al.* suggest to alter the encoding of attributes values and work with capital and small letters of secretly selected attributes [39]. According to the bit to embed the complete word (or phrase) or only the first letter is capitalized. Notice that even if their authors present these methods as being robust, the watermark will be easily identified and erased by means of simple manipulations changing the way the attributes are encoded (e.g., fixing the number of spaces or changing words' capitalization).

155124	Inflammation of right knee

155124	Inflammation of right knee

Figure 2. An example of distortion-free watermarking of categorical attributes considering the method by Al-Haj and Odeh [37].

4.2. Fragile methods

In contrast to robust methods, fragile methods are designed so as to make the watermark disappear after database manipulations. This makes them of interest for verifying the integrity of a database (tamper detection), which is the main objective of the following methods.

4.2.1. Distortion-free methods

As stated in Section 4.1.3.3, by definition, such kind of methods do not modify the values of the attributes and basically consist of introducing "virtual attributes" or the modulation of the tuples' (or attributes') organization in the relation.

Prasannakumari [40] proposed the addition of one or several virtual attributes into the relation, which will contain the watermark information. They propose the following steps. In a first time, groups of tuples are constructed. One or several attributes of NULL value are next inserted in all tuples of the relation. For each group, the value of the virtual attribute is replaced by an aggregate of the values of a chosen numerical attribute in the group. The aggregate can be the sum, the mean value, the median, etc. Then, for each tuple, the checksum of each attribute is calculated and concatenated to the virtual attribute value. At the verification stage, the same steps are followed. Integrity of data is verified if recomputed checksums correspond to the extracted ones. Working at the tuple level allows the identification of the records of the database that have been modified.

Regarding "attribute distortion-free" strategies playing with the tuple organization, that is to say reorganizing the way tuples are ordered in the database, they have been originally introduced by Li *et al.* [11]. Their scheme works as follows. In order to embed the watermark, tuples are grouped and ordered into a group depending on the value of a hash function calculated on some attributes concatenated with the tuple primary key and the owner secret key. Database is then rewritten or reorganized so as to store tuples according to the increasing order of their hash. For a group i, the watermark is a sequence W_i of length $l_i = N_i/2$ with N_i the number of tuples in the group. Insertion consists of reorganizing the order of pairs of tuples in the group depending on the bit to embed. One bit of W_i is embedded in a pair by interchanging the position of the tuples in the database so as to encode "1" or left unchanged in order to encode "0". Thus at the detection, if in a pair the hashes of the tuples do not respect their hash ordering, a bit value "1" will be extracted, "0" on the contrary. Other approaches that have been proposed later on allow the identification of the manipulations the database underwent (e.g., tuple suppression) [12, 41] or the increase of the embedding capacity, that is, the number of watermark bits that can be embedded [13].

http://dx.doi.org/10.5772/intechopen.68412

It is important to notice that methods based on tuple reordering are extremely fragile and constrain database handling by the database management system (DBMS). Tuple order should be preserved. As a consequence, their application context remains limited.

4.2.2. Lossless or reversible methods

Lossless watermarking is well adapted for verifying the integrity or authenticity of a database. For instance, it allows the embedding of a digital signature or a message authentication code like SHA [15] computed over the whole database. At the verification stage, one just has to extract the watermark, restore the database and compare the extracted signature with the recomputed one. If signatures do not match, the database has been modified. Such protection is based on fragile lossless watermarking and different strategies have been proposed. Again, these methods have been derived from reversible watermarking scheme or modulation proposed for images.

4.2.2.1. Methods working on numerical data

The histogram shifting (HS) modulation, a well-known lossless modulation for images, has been applied by Zhang *et al.* [14] to partial errors in a relation, that is in the differences between the values of one attribute of two consecutive tuples. The histogram of one digit of these differences is computed. Classes on the right side of the maximum class of the histogram are shifted to the right, creating thus an empty class close to the histogram maximum (see **Figure 3**). The attributes, the value of which belong to the maximum class, are then shifted to empty class value so as to code a "1" or left unchanged to code "0." The main drawback of this approach stands on the fact that consecutive tuples in the relation are not necessarily correlated values (contrary to contiguous pixels in an image). As a consequence, the less significant digits of the calculated differences may follow an almost uniform distribution, which seriously reduces

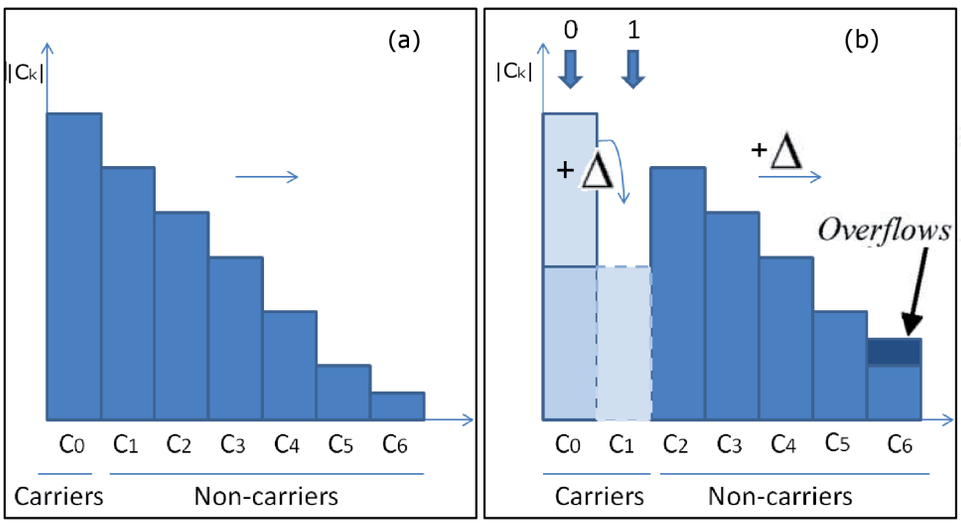

Figure 3. Histogram shifting modulation. (a) Original histogram and (b) histogram of the watermarked data.

the embedding capacity. Notice that in an image, the pixel difference distribution is close to a Gaussian distribution of small standard deviation.

Another approach proposed by Chang and Wu [42] considers the use of a support vector machine (SVM) classifier. One SVM is trained with a set of tuples selected so as to obtain a classification function $f(V)$ used by the next to predict the values of one numerical attribute. Then, they apply the difference expansion modulation, another well-known lossless watermarking modulation, for message embedding. Basically, this modulation expands the differences between original and predicted values adding one virtual least significant bit that is used for message embedding. The distortion magnitude is unpredictable and, as underlined by its authors, it can be high in some cases.

4.2.2.2. Methods working on categorical data

Coatrieux *et al.* [15] adapted the histogram shifting modulation to categorical data, being the first lossless watermarking method for *this* kind of attributes. Following the general watermarking chain exposed in **Figure 1**, one group of tuples is secretly divided in two subgroups, SG_1 and SG_2. The number of occurrences of each value of the attributes considered for embedding in SG_1 is used to construct a virtual dynamic. More clearly, values are organized depending on their cardinality, as exposed in **Figure 4**. Attributes of SG_2 are then watermarked accordingly to this virtual dynamic based on histogram shifting modulation (see **Figure 3**). The embedded watermark can be a signature of the database used to verify its integrity.

4.3. Comparative view of database watermarking methods

A synthetic classification of the methods described in Sections 4.1 and 4.2 is given in **Table 1**, depending on: the type of the watermarking modulation, the applicative context, the type of the watermarked data, the type of watermark distortion control and, finally, the robustness or fragility of the watermark against common attacks.

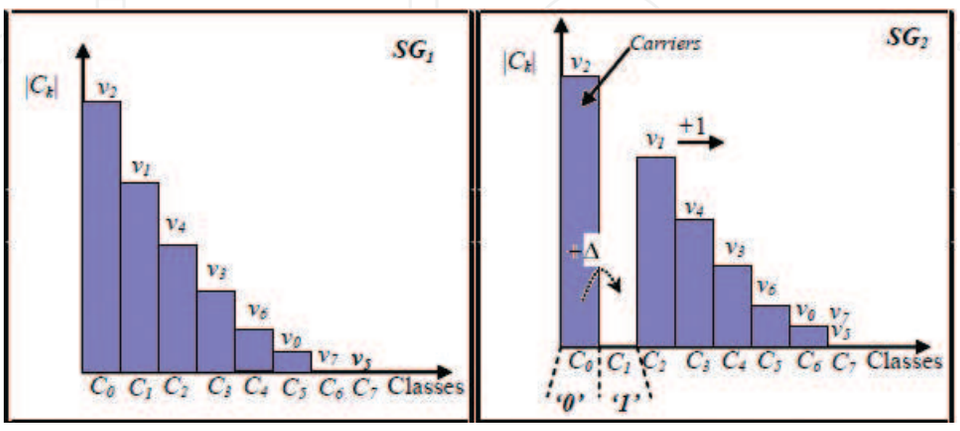

Figure 4. Histogram shifting applied to one categorical attribute with values $\{c_i\}_{i=0,\ldots,7}$. SG_1- tuple subgroup use to derive the histogram x-axis (i.e. "virtual dynamic"). $\{c_i\}$ are sorted depending of their occurrences in SG_1. SG_2 – tuple subgroup use for embedding applying HS based on SG_1 virtual dynamic.

Authors	Method	Application	Data type	Distortion control	Robustness
Agrawal and Kiernan [5]	LSB substitution	Ownership proof	Numerical	No	Robust
Li et al. [8]	LSB substitution	Traitor tracing	Numerical	No	Robust
Wang et al. [20]	Insertion of an image into LSB	Ownership proof	Numerical	No	Robust
Wang et al. [21]	Insertion of speech into LSB	Ownership proof	Numerical	No	Robust
Sion et al. [22]	MSB of the frequency histogram	Ownership proof	Categorical	No	Robust
Pournaghshband [23]	Insertion of fake tuples	Ownership proof	–	No	Robust
Lafaye et al. [26]	Modification of pairs of tuples	Traitor tracing	Numerical	Yes, respecting query constraints	Robust
Sion et al. [17]	Modification of statistics	Ownership proof	Numerical	Yes, rollback if required	Robust
Shehab et al. [18]	Modification of statistics	Ownership proof	Numerical	Yes, controlled by optimization techniques	Robust
Kamran et al [27]	Modification of LSB	Ownership proof	Numerical	Yes, parameters respecting constraints	Robust
Kamran et al [28]	Modification of LSB	Ownership proof	Numerical	Yes, controlled by optimization techniques	Robust
Gupta and Pieprzyk [31]	Difference expansion	Ownership proof	Numerical	Yes, reversible	Robust
Farfoura et al. [32]	Insertion into the fractional part	Ownership proof	Numerical	Yes, reversible	Robust
Li et al. [34]	Polar angle expansion	Ownership proof	Numerical	Yes, reversible	Robust
Franco Contreras et al. [35]	Circular histogram shifting	Ownership proof	Numerical	Yes, reversible	Robust
Chang et al. [19]	Modification of letters	Ownership proof	Categorical	Yes, reversible	Robust
Al-Haj and Odeh [37]	Modification of spaces between words	Ownership proof	Text	No but distortion has no impact	Robust
Haryunwinfura et al. [38]	Modification of spaces between words	Ownership proof	Text	No but distortion has no impact	Robust
Shah et al. [39]	Capital/noncapital letters	Ownership proof	Text	No but distortion has no impact	Robust
Prasannakumari [40]	Insertion of attributes	Tamper detection	–	Distortion free	Fragile

Authors	Method	Application	Data type	Distortion control	Robustness
Li et al. [11]	Reordering of order	Tamper detection	–	Distortion free	Fragile
Kamel and Kamel [12]	Reordering of tuples	Tamper detection	–	Distortion free	Fragile
Bhattacharya and Cortesi [41]	Reordering of tuples	Tamper detection	–	Distortion free	Fragile
Guo [13]	Reordering of tuples	Tamper detection	–	Distortion free	Fragile
Zhang et al. [14]	Histogram shifting of partial errors histogram	Tamper detection	Numerical	Yes, reversible	Fragile
Chang and Wu [42]	SVM used to predict values in detection	Tamper detection	Numerical	Yes	Fragile
Coatrieux et al. [15]	Histogram shifting of categorical attributes	Tamper detection	Categorical	Yes, reversible	Fragile

Table 1. A synthetic overview of database watermarking methods.

5. Preserving semantic data quality in database watermarking

As exposed in the previous section, distortion control-based watermarking methods mainly focus on minimizing the distortion of the database statistics. Such consideration does not necessarily take into account the semantic relationships that exist in between the attributes' values in a tuple. Database semantics should not be neglected; it will avoid the introduction of incoherent or impossible records by the watermarking process, records that give clues about the presence of a watermark to an attacker.

In this section, we propose the use of an ontological model of the semantic links in between the attributes' values in a relational database in order to minimize the distortion [29].

5.1. Definition and main components of ontology

By definitions in the literature [43–45], ontologies allow defining shared concepts in some specific area of knowledge and how these are related by means of a common vocabulary in order to overcome the intrinsic heterogeneity and complexity of the real world. An important feature of ontologies is that they are interpretable by both human operators and computer programs, representing a gateway between human and artificial knowledge.

Even though authors do not come to an agreement in terms of what components ontology should have, most definitions contain the following elements defined by Gruber [45]: classes, relations, axioms and instances.

Concepts or classes correspond to abstract groups, sets or collections of objects. Examples of concepts could be: *Person, Car, Thing*, etc. The notion of classes depends on ontology.

http://dx.doi.org/10.5772/intechopen.68412

For instance, one can define the class "*Thing*" that (in the abstract sense of the word) may contain anything one could imagine (e.g., *Person, Car, Book*, etc.). As exposed, classes can contain other classes and a universal class may contain every other class.

An individual or instance corresponds to the ground level concept of ontology; it is a concrete instantiation of an element or an object (e.g., a person named *Peter* or a car *Renault Clio*). Notice that the frontier between an individual and a class is quite blurred. It relies on the considered ontology. Individuals are described in ontology by a set of attributes. Examples of attributes can be *has-name*, *has-age* and so on. The value of an attribute is defined by a data type, for example, integer, string.

Objects in the domain are associated by means of relations specifying interactions between them. We can have relations between classes, between an individual and a class, between individuals, etc. For example, we know that one person *is-child-of* another person or that Batman *fights-against* the Joker.

We invite the reader to consult [43–45] for more information about ontological modelling of knowledge.

5.2. Relational databases and ontologies

A relational database consists of a finite set of relations $\{R_i\}_{i=1,\ldots,NR}$ where one relation R_i contains a set of N unordered tuples $\{t_u\}_{u=1,\ldots,N}$, each of which having M attributes $\{A_1, A_2, \ldots, A_M\}$ (see Section 3). Defined as such, this data structure lacks semantic information about the attributes meaning and links between different attributes' values in a tuple. Considering as an example the database of inpatient stay records, with a relation to the records which include the attributes *Gender and Diagnosis*, one should take care that the watermarking process does not turn a record such as {*Gender* ="*female*", *Diagnosis* ="*pregnant*"} into { *Gender* ="*male*", *Diagnosis* ="*pregnant*"}.

To overcome this issue, we propose to use the concepts and relations of ontology associated with the database in order first to model the knowledge one can have of the semantic relationships in between attributes of tuples and second to identify the maximum tolerated distortion for numerical attributes that will be watermarked. To do so, one question to consider is how to *make* interact these two database and ontology structures.

As stated above, concepts in an ontology are linked by means of relationships that specify hierarchical or associative interactions between them. From this standpoint, each domain value, subset or a range of values of an attribute A_t can be associated with one ontology concept. We depict in **Figure 5** an illustrative extract of such mapping considering the example of one database containing pieces of information related to in-patient stay records and its associated ontology, ontology one must *a priori* know or elaborate. Notice that such relations cannot be easily identified by means of a simple statistical analysis of the database.

In this example, the value "*Alzheimer*" in the domain of the attribute "*diagnosis*" can be associated to a concept "*Alzheimer*" in medical ontology. This concept is related to another concept "*≥ 60 years old*," which can be mapped into a range of possible values for the attribute "age."

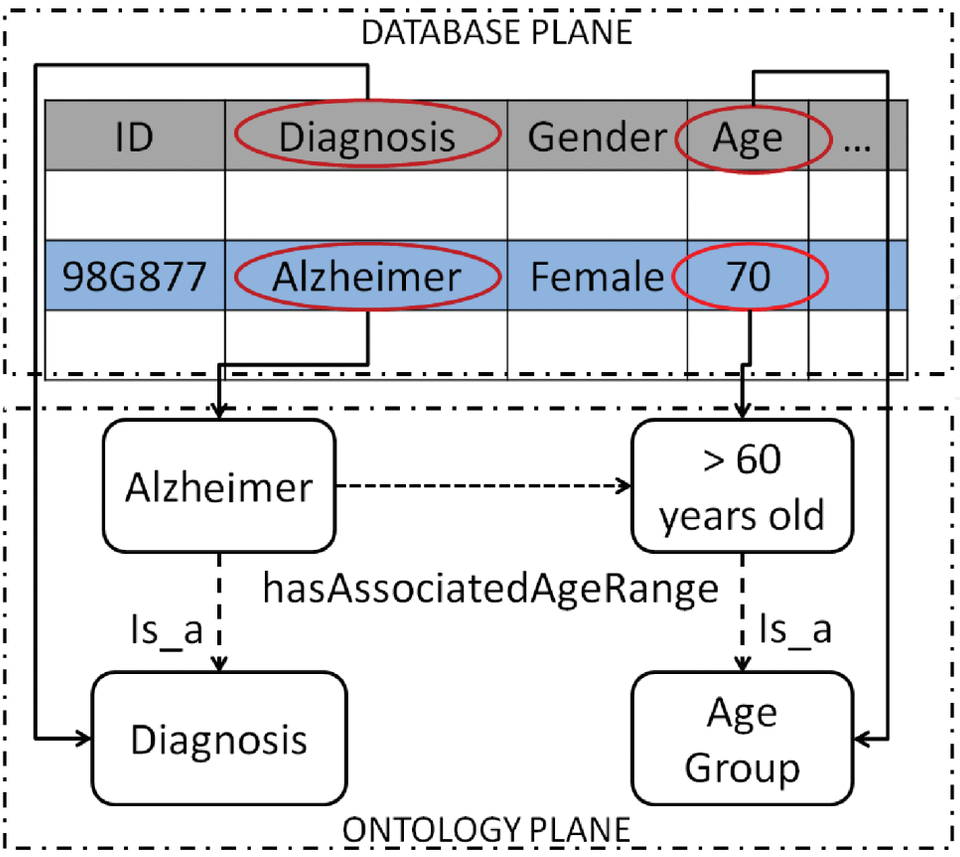

Figure 5. An existing connection between a relational database (database plane) and ontology (ontology plane) in the case of a database of in-patient stay records. Dotted and dashed arrows represent ontological relations between concepts in the ontology. Solid arrows represent connections between attributes or attributes values and ontological concepts [29].

From a watermarking point of view, this semantic relationship informs us that one attribute age value should not be turned into a value smaller than 60 in a tuple where the "*diagnosis*" attribute value is "*Alzheimer*." If we generalize, assuming the numerical attribute A_t is considered for watermarking, its value in the u^{th} tuple, that is $t_u.A_t$, semantically depends on the set of values $S_{tu.At}$ of the other attributes of t_u, that is $t_u.\{A_1, \ldots, A_{t-1}, A_{t+1}, \ldots, A_M\}$ or a subset of them.

The distortion limits of $t_u.A_t$ can be defined as $Rg_{tu.At,}$ that is, the allowable range of values $t_u.A_t$ can take after the watermarking process under the semantic constraint set $S_{tu.At}$ in order not to introduce incoherent or unlikely tuples in the watermarked database (see **Figure 6**). If we come back to the previous example, where A_t = "*age*" is an integer, the value $t_u.age$ belongs to an integer range $Rg_{tu.age}$ imposed by the set $S_{tu.age}$ = "*Alzheimer*."

Let us now consider that a categorical attribute A_c has been selected for message embedding in the relational database DB. We recall that $t_u.A_c$ corresponds to the value of the attribute A_c in the uth tuple.

As above, the range of values $Rg_{tu.Ac}$ that $t_u.A_c$ can take is semantically linked as $S_{tu.Ac}$. Because A_c is a categorical attribute, $Rg_{tu.Ac}$ is a set of categorical values $Rg_{tu.Ac} = Val_1, \ldots, Val_{Nvals}$. Again, $Rg_{tu.Ac}$ can be identified by querying the ontology. For example, if we consider A_c = "*diagnosis*" in regard to the attribute A_{c+1} = "*gender,*" we know that for a tuple where $tu.A_{c+1}$ = "*Male,*" $t_u.A_c$ cannot be equal to "*Multiple gestation.*"

http://dx.doi.org/10.5772/intechopen.68412

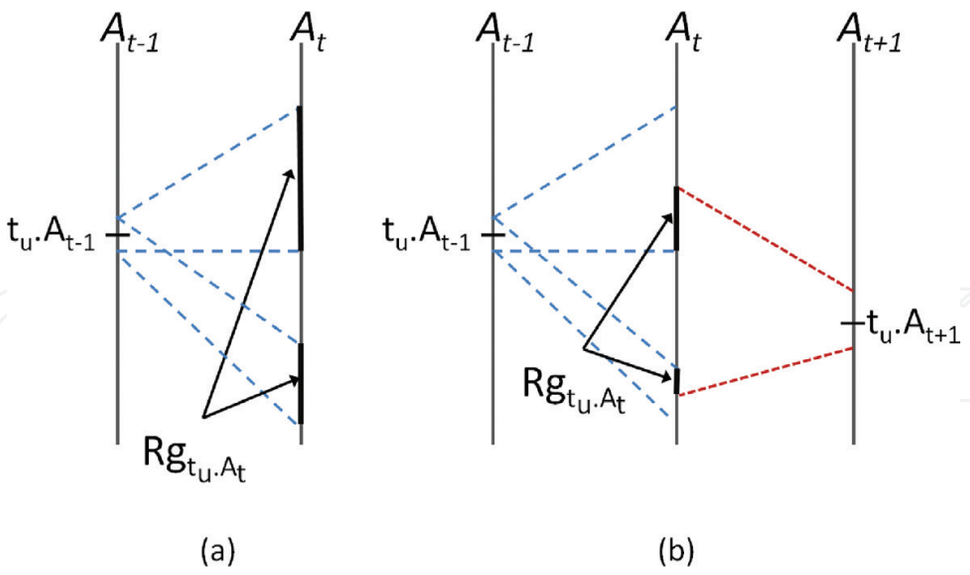

Figure 6. Identification of the allowable range $Rg_{tu.At}$ of values the attribute $t_u.A_t$ of the uth record can take for watermarking purpose. $Rg_{tu.At}$ is derived from the relationship in between the values of $t_u.A_t$ with those of other attributes of t_u: (a) when A_t is only related to A_{t-1} (in that case, the set of constraints is $S_{tu.At} = t_u.A_{t-1}$) and (b) when A_t is only related to A_{t-1} and A_{t+1} (in that case, the set of constraints is $S_{tu.At} = \{t_u.A_{t-1}, A_{t+1}\}$). In the first case, $Rg_{tu.At}$ corresponds to the union of different intervals while in the second it is at the intersection of allowable ranges imposed by the two values in $S_{tu.At}$.

It is important to notice that this semantic distortion control is complementary to any other statistical distortion control method. Indeed, additionally to the semantic constraints, one may aim at preserving the correlation or the mutual information between attributes. Then, a global solution associating semantic distortion control and statistics distortion control, such as the technique suggested by Kamran *et al*. [27] can be constructed. This was done in Ref. [29], where it was also shown that the semantic control does not reduce the watermark robustness performance.

6. Conclusion

In this chapter, we gave an overview of most recent database watermarking techniques as well as of the latest advances in terms of database distortion control. As for multimedia data, like images or video, this aspect is important due to the fact that: (i) watermarking should preserve data quality and should not interfere with the *a posteriori* use and interpretation of data and (ii) induced distortion should not betray the presence of a watermark. To do so, semantic and statistic distortion controls have both to be considered. As illustrated, ontology appears as a good candidate so as to model the semantic relationships *and* will help to avoid the occurrence of incoherent or unlikely tuples.

Several issues are still to be considered in database watermarking and have to be addressed. Regarding the control of the database distortion, the joint application of statistical and semantic constraints should be analyzed. Indeed, in order to minimize the risk of incorrect data interpretation and to ensure the correct result of data-mining operations both criteria must be considered. Moreover, the automation of distortion control is still a challenge, especially in terms of complexity.

Acknowledgements

This work has received a French government support granted to the CominLabs excellence laboratory and managed by the National Research Agency in the "Investing for the Future" program under reference ANR-10-LABX-07-01, and to the ANR project INSHARE, ANR-15-CE19-0024-02.

Author details

Javier Franco Contreras[1]* and Gouenou Coatrieux[1,2,3]

*Address all correspondence to: javier.francocontreras@watoo.tech

1 WaToo, Plouzané, France

2 Institut Mines-Telecom, IMT Atlantique Bretagne-Pays de la Loire, Brest, France

3 Institut national de la santé et de la recherche médicale, Laboratory of Medical Information Processing(LaTIM), Brest, France

References

[1] Allen M. HSBC and Falciani: How it happened [Internet]. 2015. Available from: http://www.swissinfo.ch/eng/swiss-leaks-scandal_hsbc-and-falciani--how-it-happened/41263376 [Accessed: 1 March, 2016]

[2] Rogers S. Wikileaks embassy cables: Download the key data and see how it breaks [Internet]. December 3, 2010. Available from: http://www.theguardian.com/news/datablog/2010/nov/29/wikileaks-cables-data [Accessed: 1 March 2016]

[3] Coatrieux G, Quantin C, Montagner J, Fassa M, Allaert F-A, Roux C. Watermarking medical images with anonymous patient identification to verify authenticity. Studies in Health Technology and Informatics. 2008;**136**:667-672

[4] Bouslimi D, Coatrieux G, Cozic M, and Roux C. A joint encryption/watermarking system for verifying the reliability of medical images. IEEE Transactions on Information Technology in Biomedicine. 2012;**16**(5):891-899

[5] Agrawal R, Kiernan J. Watermarking relational databases. In Proceedings of the 28th international conference on Very Large Data Bases (VLDB '02). VLDB Endowment. pp. 155-166

[6] Cox I, Miller M, Bloom J, Fridrich J, Kalker T. Digital Watermarking and Steganography. 2nd ed. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA; 2008

[7] Wagner N. Fingerprinting. In: Symposium on Security and Privacy. IEEE Computer Society, 1983. pp. 18-22

[8] Li Y, Swarup V and Jajodia S. Fingerprinting relational databases: Schemes and specialties. IEEE Transactions on Dependable and Secure Computing. 2005;**2**(1):34-45

[9] Boneh D, Shaw J. Collusion-secure fingerprinting for digital data. IEEE Transactions on Information Theory. 1998;**44**(5):452-465

[10] Tardos G. Optimal probabilistic fingerprint codes. In: 35th ACM Symposium on Theory of Computing. ACM; 2003. pp. 116-125

[11] Li Y, Guo H, Jajodia S. Tamper detection and localization for categorical data using fragile watermarks. In: 4th ACM workshop on digital rights management. ACM; 2004. pp. 73-82

[12] Kamel I, Kamel K. Toward protecting the integrity of relational databases. In: World Congress on Internet Security. IEEE; 2011. pp. 258-261

[13] Guo J. Fragile watermarking scheme for tamper detection of relational database. In: International Conference on Computer and Management. IEEE; 2011. pp. 1-4

[14] Zhang Y, Niu X, Yang B. Reversible watermarking for relational database authentication. Journal of Computers. 2006;**17**(2):59-66

[15] Coatrieux G, Chazard E, Beuscart R, Roux C. Lossless watermarking of categorical attributes for verifying medical data base integrity. In: Annual International Conference of the IEEE EMBS. IEEE; 2011. pp. 8195-8198

[16] Agrawal R, Haas PJ, Kiernan J. Watermarking relational data: framework, algorithms and analysis. The VLDB Journal. 2003;**12**(2):157-169

[17] Sion R, Atallah M, Prabhakar S. Rights protection for relational data. IEEE Transactions on Knowledge and Data Engineering. 2004;**16**(12):1509-1525

[18] Shehab M, Bertino E, Ghafoor A. Watermarking relational databases using optimization-based techniques. IEEE Transactions on Knowledge and Data Engineering. 2008;**20**(1):116-129

[19] Chang C-C, Nguyen T-S, Lin C-C. A blind robust reversible watermark scheme for textual relational databases with virtual primary key. In: Digital-Forensics and Watermarking, vol. 9023 of LNCS. Springer, 2015. pp. 75-89

[20] Wang C, Wang J, Zhou M, Chen G, Li D. ATBaM: An Arnold transform based method on watermarking relational data. In: International Conference on Multimedia and Ubiquitous Engineering. IEEE; 2008. pp. 263-270

[21] Wang H, Cui X, Cao Z. A speech based algorithm for watermarking relational databases. In: International Symposiums on Information Processing. IEEE. 2008, pp. 603-606

[22] Sion R, Atallah M, Prabhakar S. Rights protection for categorical data. IEEE Transactions on Knowledge and Data Engineering. 2005;**17**(7):912-926

[23] Pournaghshband V. A new watermarking approach for relational data. In: 46th Annual Southeast Regional Conference. ACM; 2008. pp. 127-131

[24] Suchanek FM, Gross-Amblard D. Adding fake facts to ontologies. In: 21st International Conference Companion on World Wide Web. ACM; 2012. pp. 421-424

[25] Gross-Amblard D. Query-preserving watermarking of relational databases and xml documents. In: Twenty-Second ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. ACM; 2003, pp. 191-201

[26] Lafaye J, Gross-Amblard D, Constantin C, Guerrouani M. Watermill: An optimized fingerprinting system for databases under constraints. IEEE Transactions on Knowledge and Data Engineering. 2008;**20**(4):532-546

[27] Kamran M, Suhail S, Farooq M. A robust, distortion minimizing technique for watermarking relational databases using once-for-all usability constraints. IEEE Transactions on Knowledge and Data Engineering. 2013;**25**(12):2694-2707

[28] Kamran M, Suhail S, Farooq M. A formal usability constraints model for watermarking of outsourced datasets. IEEE Transactions on Information Forensics and Security. 2013;**8**(6):1061-1072

[29] Franco-Contreras J, Coatrieux G. Robust watermarking of relational databases with ontology-guided distortion control. IEEE Transactions on Information Forensics and Security. 2015;**10**(9):1939-1952

[30] Coatrieux G, Pan W, Cuppens-Boulahia N, Cuppens F, Roux C. Reversible watermarking based on invariant image classification and dynamic histogram shifting. IEEE Transactions on Information Forensics and Security. 2013;**8**(1):111-120

[31] Gupta G, Pieprzyk J. Database relation watermarking resilient against secondary watermarking attacks. In: ICISS. 2009; pp. 222-236

[32] Farfoura ME, Horng S-J, Lai J-L, Run R-S, Chen R-J, Khan MK. A blind reversible method for watermarking relational databases based on a time-stamping protocol. Expert Systems with Applications. 2012;**39**(3):3185-3196

[33] Alattar A. Reversible watermark using the difference expansion of a generalized integer transform. IEEE Transactions on Image Processing. 2004;**13**(8):1147-1156

[34] Li Z, Liu J, Tao W. A novel relational database watermarking algorithm based on clustering and polar angle expansion. International Journal of Security and Its Applications. 2013;**7**(2):1-14

[35] Franco-Contreras J, Coatrieux G, Cuppens-Boulahia N, Cuppens F, Roux C. Robust lossless watermarking of relational databases based on circular histogram modulation. IEEE Transactions on Information Forensics and Security. 2014;**9**(3):397-410

[36] De Vleeschouwer C, Delaigle J-F, Macq B. Circular interpretation of bijective transformations in lossless watermarking for media asset management. IEEE Transactions on Multimedia. 2003;**5**(1):97-105

[37] Al-Haj A, Odeh A. Robust and blind watermarking of relational database systems. Journal of Computer Science. 2008;**4**(12):1024-1029

[38] Hanyurwimfura D, Liu Y, Liu Z. Text format based relational database watermarking for non-numeric data. In: International Conference on Computer Design and Applications (ICCDA). IEEE; 2010. pp. V4, 312–V4, 316

[39] Shah S, Xingming S, Ali H, Abdul M. Query preserving relational database watermarking. Informatica. 2011;**35**(3):391-397

[40] Prasannakumari V. A robust tamperproof watermarking for data integrity in relational databases. Research Journal of Information Technology. 2009;**1**(3):115-121

[41] Bhattacharya S, Cortesi A. A distortion free watermark framework for relational databases. In: International Conference on Software and Data Technologies; 2009. pp. 229-234

[42] Chang J-N, Wu H-C. Reversible fragile database watermarking technology using difference expansion based on SVR prediction. In: International Symposium on Computer, Consumer and Control. 2012; pp. 690-693

[43] Neches R, Fikes R, Finin T, Gruber T, Patil R, Senator T, Swartout WR. Enabling technology for knowledge sharing. AI Magazine. 1991;**12**(3):36-56

[44] Gomez-Perez A, Benjamins VR. Applications of ontologies and problem-solving methods. AI Magazine. 1999;**20**(1):119-123

[45] Gruber TR. A translation approach to portable ontology specifications. Knowledge Acquisition. 1993;**5**(2):199-220

IntechOpen